

LISI & Doc'INSA

Département Informatique

Sorin NECULITA

PFE 2000-2001

Conception d'une chaîne de conversion : format RTF vers XML

Rapport final

Description	
Projet	PFE 2000-2001 – Sorin NECULITA
Titre	Conception d'une chaîne de conversion des thèses RTF en format XML
Type de document	Rapport final
Version	1.0
Date	23 juillet 2001
Auteur	Sorin NECULITA
Distribution	LISI : Béatrice Rumpler Sylvie Calabretto Doc'INSA : Monique Joly Dalila Boudia Jean Michel Mermet

Sommaire

1. Remerciements ***3***

2. Le Projet ***4***

2.1 Objet **4**

2.2 Déroulement du projet **4**

2.3 Analyse des besoins **4**

3. Contexte ***6***

3.1 Les thèses soutenues à l'INSA de Lyon **6**

3.2 Intérêt de la sauvegarde en format XML **8**

4. Analyse de l'existant ***10***

4.1 Les applications de conversion **10**

a) UpCast (version 2.0) 10
b) Majix (version 1.2.1) 11
c) RTF2XML (version 0.9) 11
d) MathType (version 4) 11
e) Word (version 2000) 12
f) Le projet OpenOffice (version 619) 13
g) Le projet OpenOffice et MathML 13
h) RTF4XML 13
i) Conclusion sur les outils de conversion 15

4.2 Les DTD **17**

a) Structure d'une thèse soutenue à l'INSA de Lyon 17
b) Méta données - Dublin Core 21
c) Méta données – Groupe de travail ministériel 22
d) La DTD Open eBook 23
e) ETD – Electronic Thesis and Dissertation Initiative 25
f) DocBook 26
g) ISO 12083 XML DTDs 27
h) DTD TEI 28
i) Conclusion sur les DTD 29

5. Présentation de la maquette ***30***

5.1 Schéma de l'application **30**

5.2 Fonctionnement et résultats **30**

5.3 Points restants à faire **31**

6. Conclusion ***32***

7. Références bibliographiques ***33***

8. Annexes ***34***

Annexe 1 34
b) Annexe 2 35
c) Annexe 3 38
d) Annexe 4 39

1. Remerciements

Je remercie tout d’abord à Mme Monique JOLY, responsable de Doc’INSA, pour m’avoir accueilli dans le cadre de son service.

Je tiens à remercier aux enseignantes responsables de mon projet Mme Béatrice RUMPLER et Mme Sylvie CALABRETTO pour leurs aide.

Un grand remerciement j’adresse à Mme Dalila BOUDIA pour ses conseils et son aide dans la rédaction des rapports.

2. Le Projet

2.1 Objet

L’objectif du projet est la réalisation d’une chaîne de conversion du format RTF (Rich Text Format) en XML (eXtensible Markup Language) des thèses électroniques soutenues à l’INSA de Lyon et déposées à la bibliothèque scientifique et technique Doc’INSA. Le format XML est le format qui a été choisi pour l'archivage des thèses.
Ce rapport est divisé en deux grandes parties. Dans une première partie, nous avons analysé les outils de conversion existants sur le marché pour pouvoir choisir ceux qui pourraient être utiles dans ce projet et qui pourraient être intégrés dans une chaîne de conversion.
Dans une deuxième partie, nous avons analysé la structure logique des thèses pour pouvoir proposer un modèle de DTD (Document Type Definition) nécessaire à la définition du document thèse XML.
Ce projet est réalisé pour Doc’INSA et il se déroule dans le cadre d’une étude du laboratoire LISI du Département Informatique sur la recherche d’information dans les ressources électroniques.

2.2 Déroulement du projet

Le projet comprend les étapes suivantes :

- **La Phase d’initialisation**. Pendant cette période seront récupérés les besoins et sera défini l’organisation du projet.
- **La Phase d’analyse de l’existant**. Durant cette période seront récupérés tous les éléments nécessaires à la conception et au développement de la chaîne de conversion :
  - Etude des formats de stockage des fichiers textes : RTF et XML.
  - Etude des outils existants sur le marché pour la conversion RTF vers XML.
  - Etude de la structure des thèses et des DTD existantes ;
- **La Phase de rédaction** du cahier des charges de l’application.

La Phase de conception du prototype qui débouchera sur la réalisation d’une maquette permettant de convertir les thèses en format XML.

2.3 Analyse des besoins

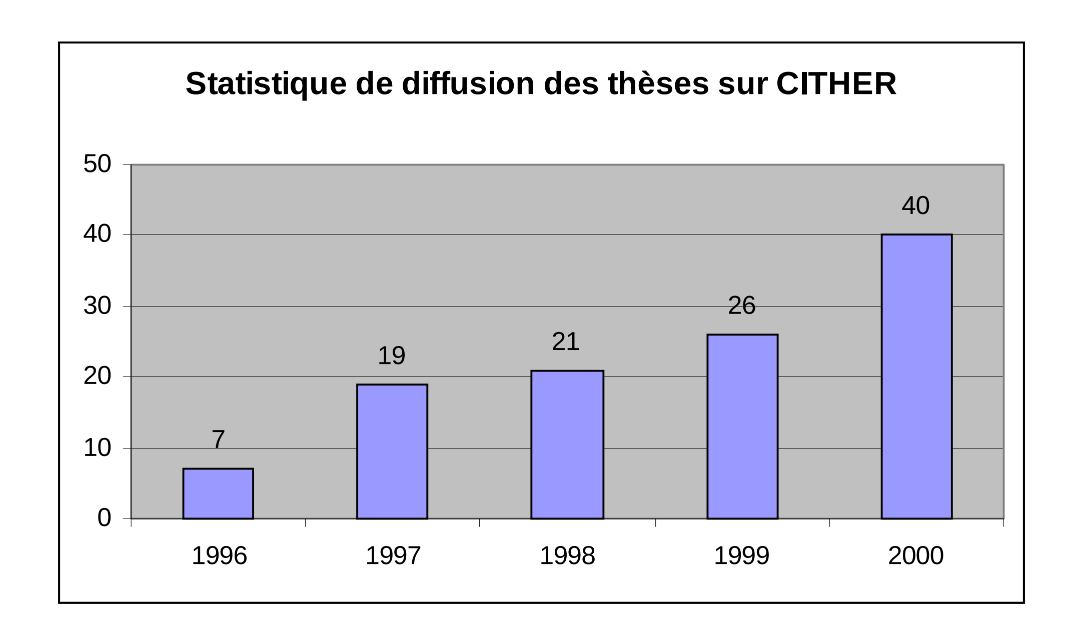
Doc’INSA reçoit les thèses au format RTF en un seul ou en plusieurs fichiers. L’objectif est de convertir tous ces fichiers en format XML et de les concaténer pour n’avoir qu’un seul document final : la thèse en format XML.
La conversion doit être complète, aucune information ne doit être perdue, qu'il s’agisse du contenu ou du style. Il faudra obtenir (en plus de la thèse en format XML) le fichier CSS (Cascading Style Sheets) contenant le style du document. Le style et le contenu seront différenciés et enregistrés dans deux fichiers distincts.

Les éléments non textuels (images, …) qui ne peuvent pas faire partie du document XML seront enregistrés sous format binaire dans des fichiers externes. Le document XML contiendra des pointeurs vers ces fichiers externes.
L'application sera développée pour les plate-formes Windows NT.
Il faudra convertir les équations éditées avec « Equation Editor 3.0 » en format MathML de présentation.

3. Contexte

3.1 Les thèses soutenues à l'INSA de Lyon

Chaque année à l'INSA de Lyon sont soutenues environ 120 thèses. Ces thèses sont déposées sous format papier à Doc'INSA (consultation et prêt).
Depuis 1997 Doc'INSA propose aux doctorants de diffuser leurs thèses sur internet. Pour cela, ils doivent fournir leurs thèses sous format électronique avec une autorisation de diffusion (Contrat à fin de diffusion d'un travail universitaire). Les formats acceptés sont le RTF et le LATEX (95% des thèses sont fournies sous format RTF, le reste de 5%, sous format LATEX). Ces thèses sont diffusées sur le WEB, sur le site CITHER (Consultation en texte Intégral des THèses En Réseau) à l'adresse électronique suivante : http://csidoc.insa-lyon.fr/these/index.html.



Chaque thèse dispose sur le site d'une page d'entrée en format HTML, appelée pont d'embarquement. Cette page est structurée sous la forme d'un sommaire, avec des liens hypertextes vers les fichiers contenant les chapitres de la thèse. Ces fichiers sont proposés sous le format PDF.
Le pont d'embarquement contient, en plus, les métadonnées de la thèses. Ces métadonnées sont cachées et décrivent la thèse (nom, prénom, titre, directeur, mots clès, résumés en français et en anglais …). Les moteurs de recherche, comme le moteur Altavista, utilise ces métadonnées pour indexer les pages web. Une fois indexées ces pages peuvent êtres retrouvées en faisant une recherche sur Rocad.

Actuellement, il existe une chaîne de conversion qui permet la transformation des formats RTF et LATEX en format PDF. Les fichiers PDF produits comportent des liens hypertextes. Ceux-ci sont générés automatiquement d'après les informations contenues dans les fichiers sources (styles…). Ils pointent sur les chapitres, sections, sous sections de la thèse et sur les fichiers qui la composent.

L'archivage se fait sous le format RTF, LATEX et PDF. Le but est de réaliser l'archivage sous le format XML car ce format dispose de plusieurs avantages (voir chapitre suivant) parmi lesquels le principal est la pérennité.

3.2 Intérêt de la sauvegarde en format XML

Les thèses seront archivées en format XML. Nous nous interessons aussi au langage MathML car celui-ci permet la conversion en format texte des équations mathématiques éditées dans MSWord.
Le langage XML (eXtensible Markup Language) est un langage de structuration des documents, relativement nouveau, créé par le groupe de travail international W3C (World Wide Web Consortium). XML est très proche de deux autres langages décrivant la structure des documents : SGML et HTML. Un des objectifs visé par le groupe W3C a été de créer un langage qui soit plus facile à utiliser que le SGML et en même temps qui puisse combler les lacunes du langage HTML.
Un document XML contient des données et des balises. Les balises marquent la présence des données. La puissance du XML par rapport au HTML est la possibilité de personnaliser les balises. La structuration des balises et leur ordre de placement dans le document sont définis dans un document à part, appelée la DTD (Document Type Definition). La DTD permet de valider la structure des documents XML.

Les avantages de la sauvegarde des documents en format XML sont les suivants :

- tout d'abord, le format XML est un format ouvert, non propriétaire, sa spécification est publique et son implémentation facile ;
- XML est un format pivot, un format d'échange entre les diverses applications de traitement de données existants à l'heure actuelle ;
- un document XML est pérenne, car il est enregistré en format ASCII (ou UNICODE) ce qui rend sa lecture et sa compréhension facile pour l'utilisateur. En plus, le document contient, au même endroit, les données et les méta données (les balises) qui décrivent son contenu. Ainsi, un utilisateur peut rapidement comprendre un document XML, sans l'aide d'un parseur ;
- la création des programmes de lecture et de traitement des documents XML est facile car d'une part, le groupe W3C a élaboré la spécification de l'interface des applications traitant des fichiers XML? (les interfaces DOM Document Object Model et SAX Simple API for XML), et d'autre part, il existe des librairies de fonctions basées sur ces spécifications et distribuées gratuitement, voire même en open source (ex : le projet XML Apache) ;
- XML tend à remplacer le langage HTML sur l'Internet car il le dépasse en possibilités d'utilisation – facilité de navigation, de recherche dans les documents, d'organisation de la présentation et de la mise en page, … (on voit déjà apparaître, avec les derniers navigateurs Internet des pages Web entièrement développées en XML);
- XML est adaptable, extensible, il peut être transformé selon les besoins, selon le type des données à traiter ;

Dans notre projet de conversion des thèses scientifiques, nous avons apporté un grand intérêt au MathML.
MathML est un langage XML qui décrit les formules mathématiques. Une question importante a été la modalité de conversion en MathML des équations (Equation Editor 3.02) éditées dans les documents RTF.
Les intérêts de la sauvegarde en MathML sont les suivants :

- MathML tend à devenir le langage universel de description des formules mathématiques ;

- MathML permet de réutiliser les équations (si une équation est enregistrée en format image, elle n'est pas accessible à certaines applications)

D'un autre coté, le langage MathML n'a pas été reçu avec satisfaction par la communauté scientifique. D'une part, ce langage est verbeux et lourd et d'autre part il est très difficile de convertir les formats mathématiques existants (Mathematica, Equation Editor) dans ce nouveau langage.
En plus, un document MathML est incompréhensible par un utilisateur sans l'aide d'un parseur.

4. Analyse de l'existant

Dans ce chapitre, nous allons étudier les outils de conversion du format RTF vers le format XML et les DTD. Cette étude nous permettra de sélectionner les outils les plus appropriés au projet de conversion de thèses du format RTF au format XML.

4.1 Les applications de conversion

Nous avons analysé les applications suivantes pour réaliser la conversion des documents RTF en format XML : « UpCast », « Majix », « RTF2XML », « WORD », « OpenOffice », « RTF4XML » et l'outil « MathType » pour la conversion des équations éditées avec « Equation Editor » en MathML.
Pour chaque outil, nous passons en revue les fonctionnalités implémentées et les points manquants.
A la fin de ce chapitre une synthèse des outils étudiés met en évidence les éléments d'un document RTF qui posent des problèmes lors de la conversion en XML.

a) UpCast (version 2.0)

L'application convertit les fichiers RTF 1.6 en format XML 1.0. Elle produit un fichier XML avec le contenu du document et le fichier CSS contenant le style. L'application permet la récupération des styles définis par défaut (« Normal », « Heading 1 », etc…) et des styles définis par l'utilisateur.

Les images insérées dans le document sont récupérées et sauvegardées sous format WML (Wireless Markup Language). Il est possible d'enregistrer les images en format JPEG et de paramétrer le facteur de compression.

L'application est fournie en trois versions :

- **UpCastSingle** : permet d'effectuer la conversion d'un seul fichier à la fois.
- **UpCastEntreprise** : est utile pour les conversions en masse d'un grand nombre de documents. La liste des fichiers est initialisée dans un fichier de "batch". L'application lit les fichiers RTF et ensuite les convertit les uns après les autres.
- **UpCastServer** : fournit une interface de programmation (une API) qui peut être intégrée dans un code source (Java ou C). Toutes les fonctionnalités des versions Single et Entreprise sont présentes dans cette API.

Les incovenients de cette applications pour notre projet sont la non conversion des dessins et des équations en MathML.

b) *Majix (version 1.2.1)*

Cette application dispose d'une interface graphique et d'un accès en mode ligne de commande. Elle convertir directement des fichiers DOC. Elle récupère le style prédéfini dans word et le style de caractères des éléments.
Ce logiciel ne convertit pas :

- les dessins ;
- les équations.

Les images sont récupérés en format WMF.

c) *RTF2XML (version 0.9)*

Cette application est un script interprété par l'outil « Omnimark ». Le script est composé d'un ensemble de fichiers définissant une suite de règles de conversion.
Les éléments récupérés :

- En-têtes et pieds de pages ;
- tableaux ;
- listes ;
- images liées ou intégrées dans un document ;
- dessins – seulement les zones de texte ;
- styles.

Les éléments non récupérés :

- les dessins (sauf les zones de texte) ;
- les équations « Equation Editor » – conversion en image.

Caractéristiques de conversion :

- les images sont récupérées en fichier externe sous format non compressé WMF.
- le style est récupéré comme attribut de la balise <p> (paragraphe) :
  ex :
  <p stylename="header" fontsize="20"><string fontsize="20">INSA de Lyon</string></p>
  Le style est intégré dans le fichier XML et non dans une CSS à part.

L'application est intéréssante pour notre chaîne de conversion car elle permet de récupérer la plus-part des éléments d'un document RTF. En plus on peut facilement l'adapter en rajoutant de nouvelles règles de conversion.

d) *MathType (version 4)*

« MathType » est la nouvelle version du logiciel « Editeur d'Equations » édité par Design Science .
« MathType » dispose d'un outil de conversion des champs d'équations en plusieurs formats, parmi lesquels MathML. Il y a 4 versions de convertisseur MathML, chacune est adaptée à un navigateur html-xml (Amaya …).

La conversion est réalisée par l'interprétation d'un fichier texte qui contient une suite de règles de conversion. L'application est adaptable, il est possible de modifier ces fichiers et d'ajouter nos propres règles de conversion.

La conversion peut être effectuée :

- à partir de Word (en utilisant un menu MathType) – cette conversion débouche sur un document contenant tous les champs d'équations du document original convertis en MathML;
- à partir d'une API (donc récupération des fonctionnalités MathType dans le code d'une autre application).

e) Word (version 2000)

Word permet la conversion d'un document DOC dans le format XHTML. Le fichier obtenu respecte les normes XML, mais il est adapté à l'affichage dans le navigateur « MS Explorer ». Le style du document peut être récupéré dans un fichier CSS.
Les éléments récupérés sont :

- tableaux ;
- listes ;
- images ;
- les objets dessinés.

Les éléments non récupérés sont :

- en-têtes et pieds de page ;
- équations « Equation Editor 3.0 ».

Le principal intérêt de cet outil est la possibilité de récupérer les dessins (les objets "Shapes"). Ces dessins sont convertis en VML Vector Markup Language (langage interprété par les dernières versions de MS Explorer 4.5 et plus) et, en même temps, ils sont enregistrés en format GIF.
Les équations sont également récupérées sous format GIF.

Cette conversion n'est possible que si le filtre HTML (2.0) pour Word 2000 est installé.

f) Le projet OpenOffice (version 619)

OpenOffice est une projet « open source » soutenu par la société « Sun Microsystems ». OpenOffice est basé sur le code source de l'application « StarOffice » rendu publique par « Sun » ?.
OpenOffice permet d'enregistrer les document en XML. L'objectif de ce projet est d'utiliser XML comme format natif d'enregistrement pour tous les logiciels faisant partie de la suite bureautique OpenOffice.
Ce projet est en cours de développement (version 627 du 1 mai 2001) et la version finale n'est pas encore disponible.

g) Le projet OpenOffice et MathML

La suite « OpenOffice » dispose d'un outil intégré StarMath pour la création des équations mathématiques. A l'ouverture d'un document MSOffice, les équations crées avec le logiciel « Equation Editor » seront converties dans le format « StarMath » ou, si l'utilisateur le désire, seront maintenues dans le format d'origine.
La conversion « Equation Editor » - « StarMath » est bijective, une équation écrite en langage « StarMath » sera, au moment de l'exportation du fichier en format DOC, enregistrée en langage « Equation Editor ».

Pour notre projet, la conversion des équations en format « StarMath » est souhaitable. Cela permettra d'enregistrer une équation en format « MathML », plutôt qu'en format binaire ZIP. La conversion des équations « StarMath » en « MathML » est encore en développement. Quelques informations sur cette conversion sont disponibles sur la liste de discutions « devat xml.openoffice.org ».

h) RTF4XML

RTF4XML est un outil de conversion produit par la société d'édition « Publilog ». Ce logiciel doit permettre la récupération la plus complète des documents électroniques.

RTF4XML convertit des fichiers au format RTF en XML.

Un système client serveur permet de déposer des fichiers RTF sur le serveur et de les récupérer en XML, PDF et TeX.

Les éléments RTF pris en charge par l'outil de conversion :

- Les paragraphes et le style des paragraphes.
- Les tableaux.
- Les équations mathématiques sont converties en MathML. L'outil convertit sans problèmes les équations éditées avec « Equation Editor », par contre des problèmes peuvent apparaître avec les « Champs d'Equation ». Le résultat de la conversion de ce type de champs est un mixage de XML et MathML.

- Les images incorporées sont enregistrées en format EPSF (Encapsuled PostScript File) vectoriel ou bitmap. Les prochaines versions devraient convertir ces images en d'autres formats (TIFF, JPEG ou d'autres).
- Les notes de bas de page.
- Le style est complètement récupéré. La conversion génère deux fichiers ;un fichier contenant la feuille de style (le style que l'on retrouve dans le modèle du document RTF) et un autre fichier contenant le document avec des informations de style supplémentaires (les exceptions de présentation?).
  Le style est récupéré sous la forme d'attributs de l'élément « paragraphe ». Cela entraîne donc un post-traitement pour convertir ces attributs en noms de balises, pour permettre la validation du document par rapport à une DTD.

Les éléments qui ne sont pas encore gérés :

- Les en-têtes des pages.
- Les zones de texte.
- Les images liées.
- Les objets dessinées. Leur récupération semble assez délicate car il faut développer un interpréteur du module graphique de Word.
- Les notes.
- Le marquage des mots d'index.

L'application est intéressante car c'est la seule application de conversion du format RTF en format XML qui permet la récupération des équations « Equation Editor 3.0 » dans le format MathML.

i) Conclusion sur les outils de conversion

Les conversions sont différentes en fonction des outils utilisés. Dans ce paragraphe nous listons les éléments qui posent des problèmes lors de la conversion.
Ces éléments sont : l'en-tête et le pied de page, les images, les équations, les dessins et le style.

L'en-tête et le pied de page sont récupérés par la majorité des outils à l'exception de Word 2000 et de RTF4XML.

Les images sont récupérées par tous les outils étudiés. Ce qui diffère, c'est le format de stockage des fichiers résultants. Les images sont récupérées soit sous leur forme brute, non-compressée (format WMF), soit sous format JPEG ou GIF, formats reconnus par les navigateurs web.

Les équations ne sont récupérées que sous forme d'images JPEG. Le seul outil qui permet de convertir ces champs est « MathType ».

Les dessins posent des problèmes lors de la récupération. Dans le cas de certaines applications comme UpCast ou Majix, ils sont tout simplement ignorés.
RTF2XML arrive à détecter la présence des dessins mais se limite à marquer leur présence dans le fichier XML. Cet outil ne permet de récupérer, pour l'instant, que l'objet « Zone de texte » en entier (avec toutes ses caractéristiques graphiques : position, taille, etc…).
Word permet, par contre, de récupérer en intégralité les dessins. Ils sont doublement convertis, d'une part en langage VML (langage XML décrivant les objets graphiques), d'autre part en images GIF.

Le style pose aussi des problèmes de conversion.
UpCast permet de récupérer le style dans une CSS à part. Dans le fichier XML, tout paragraphe auquel nous avons appliqué un style sera marqué par une balise portant le nom de ce style.
Exemple : <ResumeFR>Ceci est un résumé.</ResumeFR>

RTF2XML récupère le style comme attribut de la balise paragraphe.
Exemple : <p stylename="ResumeFR" align="left" fontname="Comic Sans MS" fontsize="20" bold="on"><string fontname="Comic Sans MS" fontsize="20" bold="on">Style personnalisé</string></p>
Ceci pose des problèmes car le fichier XML est alourdi (tous ces attributs sont répétés au niveau de chaque paragraphe). En plus, comme le nom du style n'est pas récupéré en tant que balise, nous ne pouvons pas procéder à un test de validation du document XML par une DTD.

Après la conversion avec Word 2000 le style est défini au début du document XHTML.
Au début du fichier XHTML, nous avons par exemple :

```
p.ResumeFR, li.ResumeFR, div.ResumeFR
	{mso-style-name:"Resume FR";
	margin:0cm;
	margin-bottom:.0001pt;
```

```
mso-pagination:widow-orphan;
text-autospace:none;
font-size:10.0pt;
font-family:"Comic Sans MS";
mso-fareast-font-family:"Times New Roman";
mso-bidi-font-family:"Times New Roman";
font-weight:bold;}
```

Utilisation de ce style :

```
<p class=ResumeFR>Style personnalisé</p>
```

4.2 Les DTD

Cette partie du rapport est dédiée à l'étude de la structure des thèses soutenues à l'INSA de Lyon et des DTD proposées par les divers organismes de standardisation.
Les DTD analysées sont : « Open eBook », « ETD », « DocBook », « ISO 12083 XML-Book », « TEI ». Nous présentons aussi la spécification des méta données proposé par « Dublin Core ».

a) Structure d'une thèse soutenue à l'INSA de Lyon

Nous présentons dans cette partie les éléments composant la thèse (le tableau liste les éléments et les documents source et destination).
Les documents qui font partie d'une thèse sont :

- la Thèse ;
- le Formulaire ;
- le Folio Administratif ;
- l'Annonce de soutenance.

Description de la notation :
M - Meta données
F - Facultatif

Les éléments composant la Page de Titre

Sous Elément	Source	Destinataire
N° Ordre (M)	Formulaire	Thèse Folio administratif
Année et date (M)	Thèse	Formulaire Folio Annonce
Titre FR (M) Sous titre FR (F) (M)	Thèse	Formulaire Folio Annonce
Formation doctorale (M)	Thèse	Folio Annonce
Ecole doctorale (M)	Thèse	
Qualité de l'auteur (F)	Thèse	
Auteur (M)	Thèse	Formulaire Folio Annonce
Jury (M)	Thèse	Folio Annonce

Laboratoire de recherche	Thèse	
Type de doctorat (M)	Thèse	
+ à ajouter		
Discipline (F)	Formulaire	Thèse
Titre ANG (F) (M) Sous Titre ANG (F) (M)	Thèse	Formulaire
Mention copyright (F) (M)	Thèse	

D'autres éléments

Liste des professeurs	Thèse	
Liste des écoles doctorales	Thèse	
Résumé FR (M) Mots clé FR (M)	Formulaire	Thèse
Résumé ANG (M) Mots clé ANG (M)	Formulaire	Thèse

Les éléments composant la partie administrative

Autorisation de diffusion par l'auteur	Formulaire	Thèse
Autorisation de reproduction	Formulaire	Thèse
Autorisation diffusion par le jury(M)	Formulaire	Thèse
Mention de correction	Formulaire	Thèse
Mention de confidentialité	Formulaire	Thèse
Date de fin confidentialité	Formulaire	Thèse

Code Bibliographique	Formulaire	Thèse
Code BIU	Folio	

Les éléments composant le corps de la thèse

Les préliminaires

Sous Elément	Source	Destinataire
Dédicace (F)	Thèse	
Remerciements (F)	Thèse	
Table de matières	Thèse	
Liste des figures (F)	Thèse	
Liste des tableaux (F)	Thèse	

Le contenu

Introduction	Thèse	
Chapitres	Thèse	
Sections (1,2,3,…)	Thèse	
Conclusion	Thèse	

Les post liminaires

Bibliographie	Thèse	
Annexes	Thèse	

Schéma de la thèse

La DTD définissant la structure d'une thèse INSA est proposée dans le tableau ci-dessous. Dans ce tableau, pour chaque élément est défini son nombre d'occurrences, le fait qu'il soit facultatif ou non et qu'il soit une méta donnée ou non

Eléments – sous éléments	Nb. occurrences	O/F	M
Thèse			
Page de Titre	1	O	
N°Ordre	1	O	M
Date	1	O	M
Titre FR	1	O	M
Sous titre FR	0-1	F	M
Titre ANG	0-1	F	M
Sous titre ANG	0-1	F	M
Discipline	0-1	F	M
Ecole Doctorale	1	O	M
Formation Doctorale	1-n	O	M
Auteur	1-n	O	M
Qualité auteur	0-1	F	
Jury	1	O	M
Président	1	O	
Membre	1-n	O	
Personne			
Fonction			
Laboratoire de recherche	1-n	O	
Copyright	0-1	F	M
Liste professeurs	1	O	
Liste écoles doctorales	1	O	
Résumé FR	1	O	M
Résumé ANG	1	O	M
Mots clé FR	1	O	M
Mots clé ANG	1	O	M
Corps de la thèse	1	O	
Dédicace	0-1	F	
Remerciements	0-1	F	
Table de matière	1	O	
Liste des figures	0-1	F	
Liste des tableaux	0-1	F	
Introduction	1	O	
Chapitre	1-n	O	
Section	0-n	F	
Conclusion	1	O	
Bibliographie	1	O	
Annexes	0-1	F	
Partie Administrative	1		
Autorisation diffusion Auteur	1		
Autorisation diffusion Jury	1		M
Autorisation reproduction	1		
Mention de correction	1		

Mention de confidentialité	1		
Code bibliographique	1		
Code BIU	1		

b) Méta données - Dublin Core

« Dublin Core Metadata Initiative » est une organisation ayant pour but la promotion d'un standard de méta données.
Les meta-données d'une ressource électronique proposées par la spécification « Dubli Core » sont les suivants :

- TITLE – le nom du document.
- CREATOR – l'auteur du document.
- SUBJECT .
- DESCRIPTION – un résumé sur le contenu du document.
- PUBLISHER – le nom de l'éditeur.
- CONTRIBUTOR – le nom des personnes ayant eu une contribution au contenu du document.
- DATE - (le format recommandé est le AAAA-MM-JJ comme spécifié dans la norme ISO 8601)
- TYPE – la nature du contenu.
- FORMAT – décrit le format physique et logiciel du document (permet de définir par exemple la taille ou la durée de la ressource). Cette méta donnée est utile pour connaître le logiciel de destination.
- IDENTIFIER – un numéro d'identification unique (ça peut être un URL, un ISBN ou autre).
- SOURCE – permet d'identifier le document parent dans lequel on retrouve cette ressource.
- LANGUAGE – définit la langue dans laquelle a été réalisé le document (pour la notation, l'organisation Dublin Core recommande la RFC1766 – deux lettres pour l'identification de la langue plus, éventuellement, deux lettres pour l'identification du pays. Ex : en-uk).
- RELATION – une référence vers une ressource liée.
- COVERAGE - déclaration d'un espace-temps concerné par le contenu du document.
- RIGHTS – information concernant les droits de copyright.

c) Méta données – Groupe de travail ministériel

La spécification des métadonnées proposée par le groupe de travail du Ministère Français de l'Education est basé sur la spécification Dublin Core.

Elément	*Commentaire*
DC.Contributor	"person" Nom, prénom du directeur de thèse "Directeur"
DC.Contributor	"person" Nom, prénom des membres du jury et rapporteurs selon leur rôle *zone à répéter autant de fois que de co-tutelles*
DC.Contributor	"org" Nom de l'établissement, composante, sous composante "Université de soutenance"
DC.Contributor	"org" Nom de l'établissement, composante, sous composante "co-tutelle" *zone à répéter autant de fois que de membres de jury*
DC.Coverage	
DC.Creator	"person" Nom, prénom de l'auteur *zone à répéter si plusieurs auteurs*
DC.Date	date de soutenance
DC.Date	date d'autorisation de diffusion de la thèse
DC.Description	Résumé français
DC.Description	Résumé anglais
DC.Description	Résumé en une autre langue
DC.Description	Table des matières de la thèse
DC.Format	ex "text/xml"
DC.Format	ex."3419 bytes"
DC.Identifier	URN de la thèse en texte intégral
DC.Identifier	No de la thèse attribué par l'université
DC.language	langue de la thèse, par défaut "fre"
DC.Publisher	"org" Université responsable de l'édition électronique de la thèse
DC.Relation	
DC.Rights	indique les modalités de diffusion de la thèse
DC.Rights	mention de copyright
DC.Source	Mention d'origine du document
DC.Subject	Mots clés français de l'auteur *(utiliser le **;** comme séparateur de mots clés)*

DC.Subject	Mots clés anglais de l'auteur *(utiliser le* **;** *comme séparateur de mots clés)*
DC.Subject	Mots clés de l'auteur dans une autre langue *(utiliser le* **;** *comme séparateur de mots clés)*
DC.Subject	Mots clés français conformes au thésaurus Rameau ou au MeSH en français *(utiliser le* **;** *comme séparateur de mots clés pour un même vocabulaire de référence.* *répéter la zone si le vocabulaire de référence est différent)*
DC.Subject	équivalent du code de classification sur le bordereau thèse ou pour un autre type de classification référencée *(utiliser le* **;** *comme séparateur de mots clés pour une même classification.* *répéter la zone si la classification de référence est différente)*
DC.Title	Titre et sous titre de la thèse en français
DC.Title	Titre et sous titre de la thèse en anglais
DC.Title	Titre et sous titre de la thèse en une autre langue que le français et l'anglais

d) La DTD Open eBook

La DTD Open eBook a été développée pour représenter le contenu du livre électronique. Cette spécification est destinée principalement aux éditeurs. Elle est un guide de structuration du contenu d'un livre et est accessible à diverses plates-formes de lecture électronique.

Un document Open eBook peut être composé de plusieurs fichiers et dispose d'une racine contenant la description de ces fichiers .

Les éléments composant la racine sont :

- PACKAGE IDENTITY – identificateur unique du document OeB.
- METADADA – les méta données (auteur, titre, etc.).
- MANIFEST – la liste des fichiers (images, sous- documents, autres) qui composent le document OeB.
- SPINE – définit l'ordre de lecture des fichiers composant le document.
- TOURS – définit un ordre de parcours des parties essentielles du document.
- GUIDE – contient les bibliographies, le sommaire, etc.

La structure d'un fichier est identique a la structure d'un document HTML :

- HTML
- HEAD
- BODY
- IMG
- P

- BR
- ….

Open eBook est basé sur le langage XML. Un système de lecture Open eBook est un processeur XML. Un document basé sur ce format a les caractéristiques suivantes :

- Il est un document XML valide.
- Il est conforme à la DTD Open eBook.
- Il sera conforme à la spécification XHTML, ce qui le rendra lisible par les navigateurs qui supporte ou supporteront la norme HTML 4.

Open eBook définit un langage de style basé sur le CSS1 et CSS2 en utilisant une sous partie des éléments définis dans ces spécifications et en rajoutant quelques éléments supplémentaires pour la gestion des en-têtes et des bas de page.
Open eBook supporte la norme « Dublin Core » pour la gestion des méta données.

e) ETD – Electronic Thesis and Dissertation Initiative

ETD est le nom attribué par l'Université « Virginia Tech Graduate School » (USA) aux thèses publiées en format électronique**.** Une ETD est une thèse dont le contenu respecte la DTD (désignée sous le nom de « ETD-ML ») conçue par le groupe de travail sur les thèses de « Virginia Tech ». La réalisation de la DTD a été faite en partant d'une analyse sur les thèses et les dissertations existantes et en analysant les règles régissant leur dépôt.

Une ETD est composée de trois parties :

- FRONT MATTER - correspond à la page de titre ainsi qu'aux pages suivantes qui précèdent le premier chapitre
- BODY MATTER – correspond au contenu du document.
- BACK MATTER- contenant les parties post liminaires.

Les composantes de la partie FRONT MATTER sont :

- TITLE – titre de la thèse.
- AUTHOR – le nom du doctorant.
- SCHOOL – le nom de l'université.
- DEGREE – le nom du Doctorat.
- MAJOR – le nom du département.
- APPROVAL NAMES – les noms des membres du jury.
- DATE OF DEFENCE – la date de la soutenance.
- CITY, STATE – le lieu de la soutenance.
- KEYWORDS – 4 à 6 mots clés permettant la classification de la thèse.
- COPYRIGHT – les informations sur le droit d'auteur.
- ABSTRACT – le résumé.
- DEDICACE – dédicace optionnelle.
- AUTHOR'S ACKNOWLEDGMENTS – remerciements (optionnel).
- TABLE OF CONTENTS – table de matières.
- LIST OF MULTIMEDIA OBJECTS – liste des objets multimedia.

La partie BODY MATTER est composée de :

- CHAPITRES.
- SECTIONS
- PARAGRAPHES
- …

La partie BACK MATTER contient :

- REFERENCES – la bibliographie.
- APPENDICES – les annexes.
- VITA – la biographie du doctorant.

f) DocBook

DocBook est une DTD conçue par le « DocBook Technical Committee », groupe de travail faisant partie de l'organisation OASIS (« Organization for the Advancement of Structured Information Standards ». Cette DTD a comme principal domaine d'application (mais sans y être limité) la structuration des livres électroniques ayant comme sujet l'informatique (documentation, tutorials, etc…).
Il existe une version SGML et une version XML de la DTD DocBook.

La DTD DocBook est composé de 5 documents :

- dbpoolx.mod - définit les objets et les éléments faisant parties d'un document.
- Dbhierx.mod – est spécialisé dans les manuels et les documentations. Ici est définie la hiérarchie des différentes parties du document.
- Dbnotnx.mod – déclare les différentes entités standards (jeux de caractères, les formats de fichier, etc.).
- Dbcentx.mod – déclare d'autres entités (comme les notations mathématiques).
- Dbgenent.mod – dans ce fichier on peut inclure les entités personnalisées.

Même si la DocBook est assez massive (la préface contient plus de 40 éléments : titre, sommaire, auteur, etc…) on ne peut pas l'utiliser pour définir le contenu d'une thèse. Il manque des champs comme : le département, l'école, le jury ….

g) ISO 12083 XML DTDs

Cette norme se propose de définir plusieurs formats de DTD pour les documents écrits en langage XML.
On retrouve quatre versions de DTD :

- XML Article DTD.
- XML Book DTD.
- XML Serial DTD – définit la structure des articles des périodiques.
- XML Math DTD – cette DTD fait partie des DTD Article et Book mais elle est aussi fournie en fichier séparé pour être utilisée avec d'autres DTD.

La DTD qui semble la plus appropriée à notre projet sur les thèses est la Book DTD.
Voici la structure simplifiée des éléments faisant partie de cette DTD.

Un document est composé de :

- FRONT – les pièces préliminaires.
- BODY – le corps du document.
- APPMAT – les annexes
- BACK – les post liminaires

La partie FRONT du document contient :

- TITLEGRP – définit le ou les titres du document.
- AUTHGRP – définit l'auteur du document : le nom, une organisation auquel l'auteur appartient, un degré, un rôle, une école, une adresse.
- DATE – une date de référence du document.
- PUBFRONT – des informations sur l'éditeur de la publication : ISBN, prix, etc…
- COPYRIGHT
- TOC – la table des matières

La partie BODY contient :

- CHAPITRES
- SECTIONS
- PARAGRAPHES
- …

La partie BACK contient :

- GLOSSARY
- INDEX
- NOTES
- VITA – la biographie de l'auteur
- AFTERWRD – la postface

Un élément utile qui manque dans la partie « front » est la liste du jury.
On retrouve des éléments marquant la présence des équations mathématiques et des images.

h) DTD TEI

« Text Encoding Initiative » [TEI] est un projet international qui a comme objectif la conception d'un guide pour l'encodage des textes sous forme électronique. Il s'agit d'un projet axé sur les textes littéraires et est développé par plusieurs universités (Oxford, Virginia, Bergen) et soutenu par des associations littéraires comme « Association for Computers and the Humanities » ou « Social Science and Humanities Research Council » du Canada.

Une DTD a été réalisée pour le langage SGML. Cette DTD décrit la plupart des formats des textes en sciences humaines. Elle est composée de plusieurs modules qui peuvent être combinés pour créer une DTD adaptée aux besoins spécifiques.
Le projet fournit une DTD obtenue par cette méthode. Cette DTD appelée TEI-Lite est une version allégée de l'original et contient les éléments essentiels .

L'en-tête de la DTD TEI contient des informations analogues à celles que l'on trouve sur la page de
titre d'un texte imprimé. Elle contient jusqu'à quatre parties :

- une description bibliographique du texte électronique.
- une description de la manière dont il a été codé.
- une description non-bibliographique du texte (le « profil » du texte).
- un historique de révision.

Le corps du document TEI comporte les éléments suivants

- FRONT - regroupe tous les éléments (en-têtes, page de titre, préfaces, dédicaces, etc.) situés avant le début du texte lui-même.
- GROUP - regroupe plusieurs textes unitaires ou groupes de textes.
- BODY - regroupe le corps entier d'un texte unitaire seul, à l'exclusion des pièces liminaires ou annexe.
- BACK - regroupe toutes les annexes qui suivent le texte principal.

La page de titre regroupe les éléments suivants :

- DOCTITLE - contient le titre d'un document, y compris tous ses constituants, tel que présenté sur une page de titre; doit être partagé en éléments `TITLEPART`.
- TITLEPART - contient une subdivision ou division du titre d'une œuvre.
- BYLINE - regroupe la mention de responsabilité principale d'une oeuvre donnée, tel que reproduite sur la page de titre ou au début ou à la fin de l'ouvrage.
- DOCAUTHOR - contient le nom de l'auteur du document, tel que présenté sur la page de titre (souvent mais pas toujours contenu dans un `<byline>`).
- DOCDATE - contient la date du document, telle que présentée (habituellement) sur la page de titre.
- DOCEDITION - contient une mention d'édition, telle que présentée sur une page de titre d'un document.

D'autres éléments faisant partie des pièces liminaires :

- FOREWORD - un texte adressé au lecteur, par l'auteur, le rédacteur ou l'éditeur, éventuellement sous forme d'une lettre.
- PREFACE
- DEDICACE
- ABSTRACT
- ACK - les remerciements
- CONTENTS - une table des matières.

Les pièces annexes sont :

- APPENDIX
- GLOSSARY
- NOTES
- BIBLIOGRAPHY - une série de références bibliographiques.
- INDEX - une série d'entrées d'index.
- COLOPHON - description à la fin du livre mentionnant où, quand, et par qui il a été imprimé; dans les livres modernes il donne souvent les détails de production et identifie les polices utilisées.

:

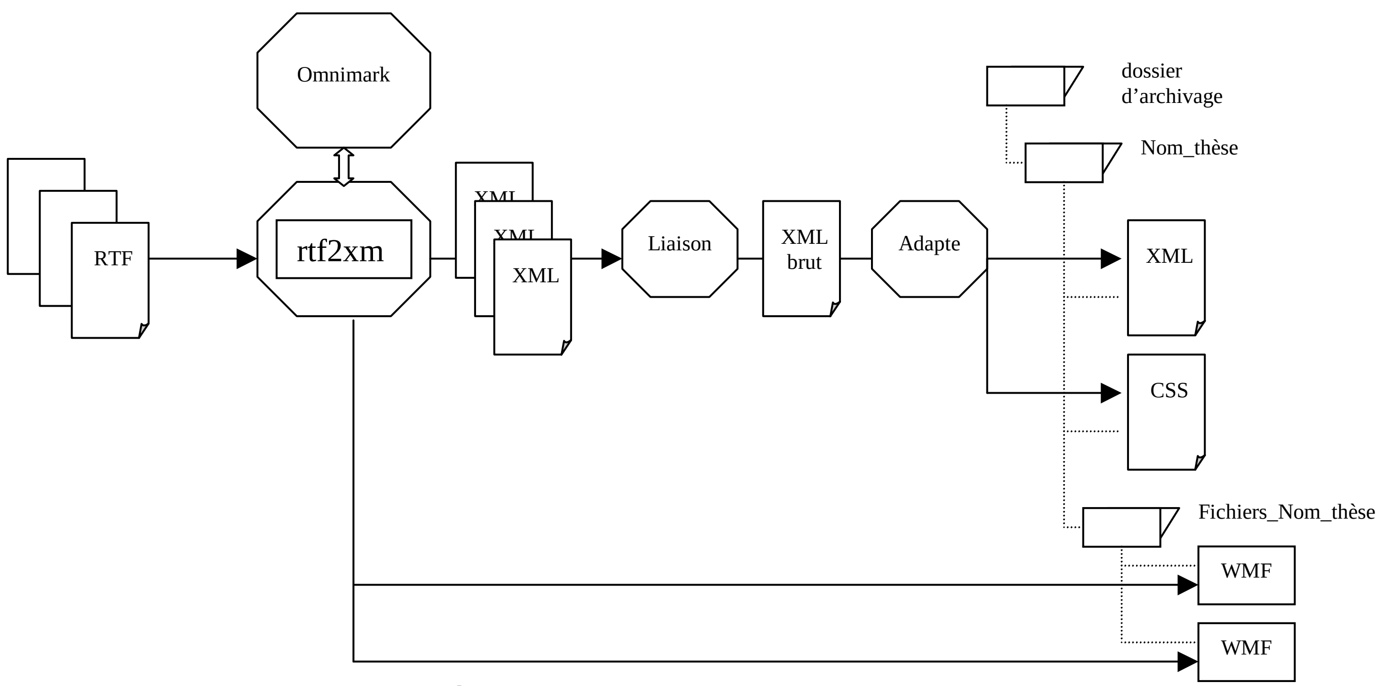
i) *Conclusion sur les DTD*

La DTD la plus intéressante et la plus proche de nos objectifs semble être la DTD « ETD » de l'université « Virginia Tech ». Elle décrit les thèses scientifiques et contient la plupart des éléments dont nous avons besoin. Restent quelques éléments qui sont spécifiques à l'INSA de Lyon (la liste des écoles doctorales, la liste des professeurs). La solution la plus évidente serait donc de reprendre la DTD « ETD » et de la compléter avec les éléments spécifiques aux thèses soutenues à l'INSA de Lyon.

5. Présentation de la maquette

La maquette et un guide de conversion sont présentés dans cette partie.

5.1 Schéma de l'application



5.2 Fonctionnement et résultats

L'application prend en entrée, le (les) fichier(s) RTF composant la thèse. Ces fichiers sont traités avec le script RTF2XML et l'outil Omnimark ce qui permet de générer des fichiers XML et d'extraire les images . Ces fichiers sont ensuite concaténés dans un seul document XML. On appelle ce document « XML brut » car sa structure ne correspond pas à nos objectifs : le style est contenu dans la structure du fichier et les noms des styles ne sont pas enregistrés en tant que noms de balises.

- On utilise donc un utilitaire qui permet de balayer le fichier XML et de créer la feuille de style CSS ;
- définir des balises portant les noms des styles.
- simplifier les balises en supprimant les redondances de style ;

Cet utilitaire a été réalisé avec la bibliothèque de fonctions Xerxes C++ implémentant la spécification SAX de parsage des documents XML **[XMLApache]**.

Supposons que le fichier XML brut contienne une ligne de la forme :
<p stylename= RESUME left=11 …>Le texte du résumé </p>
en utilisant cet utilitaire on obtient le fichier CSS :

```
<STYLE>
    .RESUME {left=11 ;…}
</STYLE>
```

et dans le fichier XML on obtient la balise :
<RESUME>Le texte du résumé </RESUME>

L'Annexe 4 contient un exemple complet de fichiers XML et CSS générés avec ce convertisseur.
L'Annexe 1 décrit la procédure d'utilisation de ce convertisseur.
L'Annexe 2 décrit le mode d'utilisation de l'utilitaire RTF2XML.
L'Annexe 3 décrit le mode d'utilisation de l'utilitaire de conversion du fichier XML brut, utilitaire qui peut être utilisé séparément de l'application principale de conversion.

5.3 Points restants à faire

Les dessins, les objets « Shapes » ne sont pas convertis. Leurs présences sont toutefois détectées et indiquées dans le fichier XML par les balises <drawing-shape>. Ceci facilite l'intégration ultérieure d'un utilitaire qui puisse reconnaître et convertir ces objets. Les balises <dawing-shape> permettront d'insérer au bon endroit les liens vers les fichiers résultant de la conversion des dessins en image.
D'autre part les équations sont enregistrées sous forme d'images WMF et non pas dans le format MathML.

6. Conclusion

Ce projet nous a permis de faire une étude sur la structure des thèses soutenues à l'INSA de Lyon pour définir la DTD à utiliser lors des conversion en format XML. En plus, nous avons pu voir l'état actuel de développement des applications qui permettent la conversion RTF vers XML.
Utilisant au mieu les possibilités actuelles, nous avons pu réaliser une maquette de chaîne de conversion. Du coté des applications, il faudrait attendre la finalisation des fonctionnalités de conversion (récupération des dessins, des équations en MathML). L'outil qui paraît le plus prometteur, de ce point de vue est « OpenOffice », dont la sortie est prévue pour l'automne 2001.

7. Références bibliographiques

[XML] Extensible Markup Language 1.0 (Second Edition) [on line]
<URL : http://www.w3c.org/TR/2000/REC-xml-20001006>

[OpenOffice] Open Office.org Source Project [on line]
<URL : http://www.openoffice.org>

[UpCast] Up-Cast [on line]
<URL: http://www.infinity-loop.de/index.html>

[XMLAppache] Projet XML Apache [on line]
<URL : http://xml.apache.org>

[Majix] Majix [on line]
<URL : http://tetrasys.dhs.org/majix.html >

[RTF4XML] RTF4XML [on line]
<URL : http://www.hcu.ox.ac.uk/TEI >

[ISOXML] ISO 12083 XML [on line]
<URL : http://www.xmlxperts.com/12083xml.htm>

[TEI] DTD TEI [on line]
<URL : http://www.hcu.ox.ac.uk/TEI>

[DocBook] DocBook [on line]
<URL : http://www.oasis-open.org/docbook/ > à revoir

[ETD] Electronic Thesis Disertation [on line]
<URL : http://csgrad.cs.vt.edu/~mbjorklu/etdml>

[OpeneBook] Open-eBook [on line]
<URL : http://openebook.org >

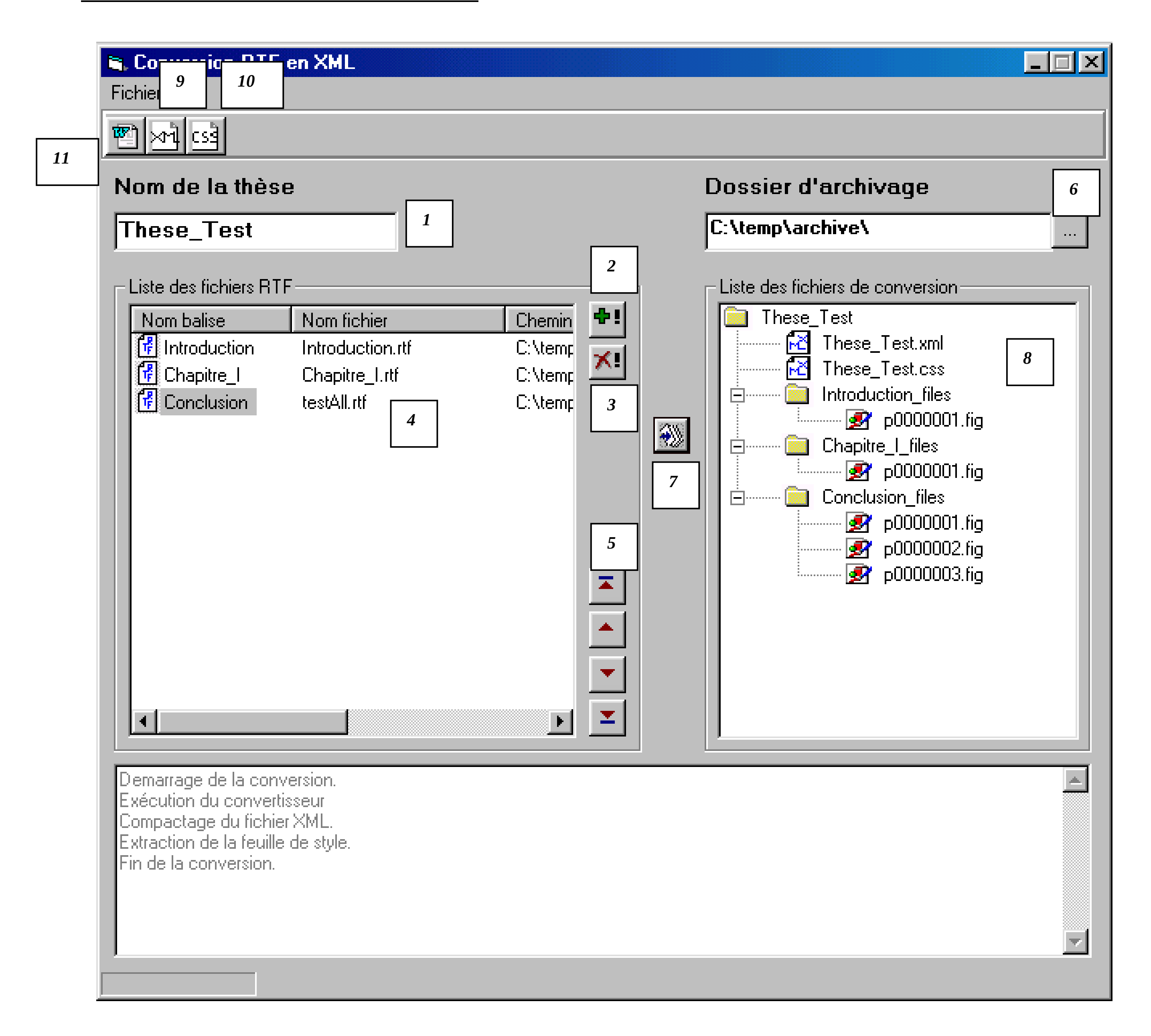
[MathType] MathType [on line]
<URL : http://www.mathtype.com/fr>

[Dublin Core] Dublin Core [on line]
<URL : http://dublincore.org >

8. Annexes

a) Annexe 1

<u>Utilisation de l'utilitaire de conversion</u>



La procédure de conversion consiste dans les étapes suivantes :

- L'utilisateur définit le nom de la thèse. Ce nom sera le nom du dossier d'archivage de la thèse en XML, ainsi que le nom du fichier XML et du fichier CSS. (1)
- L'utilisateur choisit les fichiers RTF à convertir. Il peut les ajouter (2) ou les supprimer (3) de la liste.
- La liste des fichiers RTF (4) affiche les documents que l'utilisateur a choisi de convertir. L'ordre du listing est l'ordre de concaténation des fichiers XML obtenus suite à la conversion. Cette ordre peut être changée (5).

- Dans cette liste on affiche le dossier d'emplacement du fichier RTF, le nom du fichier RTF et le nom de la balise qui marquera le contenu du document dans le fichier XML final. La valeur de la balise est, par défaut, le nom du fichier RTF. Cette valeur est directement modifiable.
- L'étape suivante consiste dans le choix du dossier d'archivage (6).
- Une fois tous ces éléments définis, l'utilisateur peut déclencher la conversion (7).
- A la fin de la conversion, on affiche dans la liste arborescente (8) le fichier XML, le fichier CSS ainsi que les fichiers annexes qui ont été générés. On peut visualiser le fichier XML (9) , le fichier CSS (10) et le fichier RTF sélectionné (11).

Limitations : les noms des dossiers, des fichiers et des balises ne doivent pas contenir des espaces et des caractères non ASCII (ex : les accents sont interdits).

La structure du fichier XML brut est définie dans les documents DTD fournis avec l'utilitaire RTF2XML. Le point d'entrée de ces documents est constitué par « transdoc.dtd ».
Le fichier XML obtenu à la fin de la conversion n'est pas relié à une DTD. Pour pouvoir le valider il faudra créer une DTD en partant du fichier « transdoc.dtd » et en rajoutant la définition des nouvelles balises.

b) Annexe 2

Utilisation de l'utilitaire RTF2XML
(extrait du manuel d'utilisation fourni avec l'utilitaire)

```
To run the RTF2XML program, you must first obtain OmniMark LE from
Omnimark Technologies at http://www.omnimark.com/. Version 4 or later is
prefered. Once you have installed OmniMark LE, then you can run this
program from the command line as follows:

omle -s rtf2xml.xom -of output.xml input.rtf


This should produce a valid XML file. (Note: you can pass multiple RTF
files as input, and they will be concatinated into a single XML file.) If
you want to use SGML instead, and your parser does not support Unicode
(i.e. if you are using OmniMark LE Version 3.x), you should run it as
follows:

omle -s rtf2xml.xom -a no-unicode -of output.xml input.rtf


or alternatively,

omle -s rtf2xml.xom -a escape-unicode -of output.xml input.rtf


Command line options:

Switches:
-a output-sgml
Outputs SGML instead of XML. This switch is an alias for the escape-unicode
option, since all it does is set escape-unicode to true, which results
in ASCII SGML as the output format.

-a no-unicode
Supresses all Unicode. Outputs alternative characters if present
in the RTF. This also results in the xml-ents and sdata-ents
```

```
parameter entities being set to IGNORE and INCLUDE respectively
in the document prolog. Note: this means that the output will be
SGML instead of XML.

-a escape-entities
Outputs an ESCAPE element instead of a named entity reference.

-a escape-ansichars
Outputs an ANSICHAR element instead of an ANSI character reference.

-a escape-unicode
Outputs a UNICODE element instead of a Unicode character reference.
Activate this if you want access to the Unicode values, but need to
process the resulting file with OmniMark V3 or any other parser that
doesn't grok UTF-8. Alternative characters will not be output except in
attribute (such as style names), but their values are accessible through
the ALT attribute of the UNICODE element. This also sets the
xml-ents and sdata-ents parameter entities to IGNORE and
INCLUDE respectively in the document prolog. Note: this means that the
output will be SGML instead of XML.

-a ansi-stylenames
Uses the ansi representation of stylenames rather than the Unicode version.

-a allow-nested-paras
By default, a FIELD is not wrapped in a paragraph when it contains
paragraphs within its FLDRSLT child. This switch allows such nested
paragraph structures.

-a link-subdocs
RTF subdocuments (different from the SGML kind) are incorporated
into the main document automatically unless this switch is set.
Activating this switch will cause all RTF subdocuments to be
referenced via the DOCLINK element.

-a extract-figures
Causes all embedded figures to be extracted from the RTF.
Note: no conversion is done. Figures are decoded from hexidecimal to
binary and placed raw on the file system.

-a extract-unlinked-figures
Extracts only those figures that do not have the "link to file"
option specified in MS-Word.

-a sdata-entities
Sets the xml-ents and sdata-ents parameter entities to IGNORE and
INCLUDE respectively in the document prolog. Note: this means that the
output will be SGML instead of XML.

-a output-drawing-objects
Outputs supported drawing objects. Currently, only textboxes and
embedded graphics that use the {/pict ...} construct are
supported.

Streams:
-d resource-path "/where_i_put_my_stuff/rtf2xml/"
Set this value if you move the RTFDOC DTD and associated files, or if you
are running RTF2XML via a shell script or batch file.

-d sgml-log "sgml.log"
This is only valuable for debugging. It spits the intermediate RTFDOC data
used in the cross-translate to the file you specify. It's useful because it
allows you to see exactly what's going to the parser.

-d fig-path "/figures/"
```

```
Specifies the directory to which embedded figures will be extracted. By
default they are extracted to the current directory.

-d fig-ext "eps"
Specifies the extension you want placed after the "." in the filename. By
default, "fig" is used.
```

Exemple de ligne de commande RTF2XML utilisée dans l'utilitaire de conversion :

```
C:\PROGRA~1\OMNIMARK\OMNIMARK.EXE -s C:\TEMP\APPLICATION\rtf2xml\rtf2xml.xom -a extract-figures -a extract-unlinked-figures -a output-drawing-objects -d fig-path
C:\temp\archive\These_Test\Introduction_files\ -of C:\TEMP\APPLICATION\temp\Introduction.xml
C:\temp\Introduction.rtf
```

c) Annexe 3

Utilisation de l'utilitaire de simplification du fichier XML et génération de la feuille de style

Cet utilitaire utilise l'interface SAX pour balayer le fichier XML brut. L'implémentation utilise la librairie Xerxes pour C++, développée dans le cadre du projet XML Apache.
L'utilisation de cet utilitaire est la suivante :

Dtdadapte fic_in.xml fic_out.css > fic_out.xml

avec :

- fic_in.xml – le fichier XML à traiter ;
- fic_out.css – le fichier CSS à générer ;
- fic_out.xml – le fichier XML à générer.

L'application génère le résultat sur la sortie standard. On utilise une redirection du flot de sortie pour écrire ce résultat dans le fichier XML.

Exemple de ligne de commande utilisée dans le convertisseur :

```
C:\TEMP\APPLICATION\\DtdAdapte\dtdadapte C:\TEMP\APPLICATION\temp\These_Test.comp.xml
C:\temp\archive\These_Test\These_Test.css > C:\temp\archive\These_Test\These_Test.xml
```

d) Annexe 4

Exemple de fichier XML généré

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<transdoc Nom_These="exemple">
<ATT00033 fileName="ATT00033.rtf" imgFilesPath=".\ATT00033_files\" charset="ANSI">
<meta>
<title>N° Ordre 99 ISAL 0089</title>
<author>Département IF</author>
<creation-date>
<year>2001</year>
<month>5</month>
<day>2</day>
<hour>15</hour>
<minute>32</minute>
</creation-date>
<revision-date>
<year>2001</year>
<month>5</month>
<day>2</day>
<hour>15</hour>
<minute>34</minute>
</revision-date>
<company>INSA de LYON</company>
<template>THESE.dot</template>
</meta>
<section>
<header page="default">

</header>
<footer page="default">
<field><fldinst> PAGE </fldinst><fldrslt>2</fldrslt></field>
</footer>
<texte11_these>N° Ordre 99 ISAL 0089<string fontsize="24">     </string>Année
1999</texte11_these>
<texte11_these></texte11_these>
<p>THESE</p>
<p>Présentée devant</p>
<Universite>L&#x2019;INSTITUT NATIONAL DES SCIENCES APPLIQUEES DE
LYON</Universite>
<p>pour obtenir</p>
<Grade>LE GRADE DE DOCTEUR</Grade>
<Formation_Doctorale>F<string scaps="on">ORMATION DOCTORALE </string>: Génie
Civil : sols, matériaux, structure, physique du bâtiment</Formation_Doctorale>
<Ecole_Doctorale><string caps="on">Ecole doctorale</string> : Mécanique, Energétique,
Génie Civil, Acoustique</Ecole_Doctorale>
<p>par</p>
```

<Auteur>Monika Woloszyn épouse Vallon</Auteur>
<Discipline>Ingénieur Génie Civil et Urbanisme</Discipline>
<Discipline>Diplômée de l'INSA de Lyon</Discipline>
<texte11_these></texte11_these>
<texte11_these></texte11_these>
<texte11_these></texte11_these>
<Titre_these>Modélisation hygro-thermo-aéraulique des bâtiments multizones</Titre_these>
<Titre_these>proposition d'une stratégie de RESOLUTION du système couplé</Titre_these>
<Titre_these></Titre_these>
<texte11_these></texte11_these>
<texte11_these></texte11_these>
<Soutenance>Soutenue le 26 novembre 1999 devant la Commission d’Examen</Soutenance>
<texte11_these></texte11_these>
<Jury>Jury : </Jury>
<Jury>MM. Jean <string caps="on">Brau</string> Président du Jury</Jury>
<Jury>Jean-Luc <string caps="on">Hubert</string> Examinateur</Jury>
<Jury>Christian INARD Rapporteur</Jury>
<Jury>Jean <string caps="on">Lebrun</string> Rapporteur</Jury>
<Jury>Gilles <string caps="on">Rusaouen</string> Directeur de Thèse</Jury>
<Jury>Stig <string caps="on">skelboe</string> Examinateur</Jury>
<Jury>Paul <string caps="on">Stangerup</string> Rapporteur</Jury>
<Jury></Jury>
<Copyright>Cette thèse a été préparée au Laboratoire CETHIL - équipe Thermique du Bâtiment de l’INSA de Lyon</Copyright>
<texte10></texte10>
<p></p>
</section>
</ATT00033>
</transdoc>

Exemple de fichier CSS généré

<STYLE TYPE="text/css">
<!--
.header{ fontsize:20; bold:on; italic:default; scaps:default; color:default; caps:default; rev-status:default; subscript:default; align:default; hidden:default; charset:0; superscript:default; rev-author:default; underline:default; fontname:default; rev-time:default; expandtwips:default; strike:default; }
.footer{ align:right; fontsize:20; italic:default; scaps:default; color:default; caps:default; rev-status:default; subscript:default; hidden:default; charset:0; bold:default; superscript:default; rev-author:default; underline:default; fontname:default; rev-time:default; expandtwips:default; strike:default; }
.texte11_these{ align:justify; fontsize:20; italic:default; scaps:default; color:default; caps:default; rev-status:default; subscript:default; hidden:default; charset:0; bold:default; superscript:default; rev-author:default; underline:default; fontname:default; rev-time:default; expandtwips:default; strike:default; }
.p{ align:center; fontsize:24; bold:on; italic:default; scaps:default; color:default; caps:default; rev-status:default; subscript:default; hidden:default; charset:0; superscript:default; rev-

author:default; underline:default; fontname:default; rev-time:default; expandtwips:default; strike:default; }
.Universite{ align:center; bold:on; italic:default; scaps:default; color:default; caps:default; rev-status:default; subscript:default; hidden:default; charset:0; superscript:default; rev-author:default; underline:default; fontname:default; fontsize:default; rev-time:default; expandtwips:default; strike:default; }
…
-->
</STYLE>